

PROJECT SUMMARY

NASA TASK ORDER 2 FINAL REPORT



TASK NO.: NNL10AB12T

Contract No.: NNL09AA10B

PROJECT TITLE:

Automated Generation of Structured CFD Grids using Topology Methods

COMPANY:

TechnoSoft Inc.
info@technosoft.com
513-985-9877

ADDRESS:

11180 Reed Hartman Highway
Cincinnati OH 45242

SUMMARY:

This report documents the work performed from March 2010 to March 2012. The Integrated Design and Engineering Analysis (IDEA) environment is a collaborative environment based on an object-oriented, multidisciplinary, distributed framework using the Adaptive Modeling Language (AML) as a framework and supporting the configuration design and parametric CFD grid generation. This report will focus on describing the work in the area of parametric CFD grid generation using novel concepts for defining the interaction between the mesh topology and the geometry in such a way as to separate the mesh topology from the geometric topology while maintaining the link between the mesh topology and the actual geometry.

SUBMITTED BY:

Hilmi N. Kamhawi

Date:

June 20, 2012



Final Report
Integrated Design Engineering Analysis (IDEA) environment
Automated Generation of Structured CFD Grids using
Topology Methods

Contract Number: NNL09AA10B / TASK NNL10AB12T

Report Prepared: June 2012

Prepared by:
TechnoSoft Inc.
11180 Reed Hartman Highway
Cincinnati OH 45242
info@technosoft.com
513-985-9877

This contract is sponsored by:
NASA Langley Research Center
Vehicle Analysis Branch
Hampton, Virginia 23681-2199

Table of Contents

Introduction.....	1
2D vehicle centerline parametric geometry	2
Low speed flow path.....	6
REST inlet.....	8
Busemann inlet.....	11
Concluding remarks	11

Introduction

The Integrated Design and Engineering Analysis (IDEA) environment is a collaborative environment based on an object-oriented, multidisciplinary, distributed framework using the Adaptive Modeling Language (AML). IDEA contains several modules supporting configuration design and parametric structured CFD grid generation. Historically, structured grid generation has been a process that was performed after a design has been finalized. Using the new techniques developed by the team, a mesh topology can be developed and used for structured mesh generation as the design evolves, allowing high fidelity CFD to be incorporated during the design process. The grid generation process relies on a novel concept where the user builds a mesh topology using geometric abstraction tools, a topology editor, and a structured mesh generator. Geometric reasoning in AML is one of the key features that facilitates the tight coupling of geometry and mesh generation.

The structured mesh can be built in a number of ways; the following is a brief discussion of the steps that are needed to build the structured mesh blocks. The user can skip some of the steps below based on the case they are working on.

1. Abstract the geometry. This can be done by selecting faces and edges from the geometry and assigning the selected entities to named properties (user defined) that are used to capture the design intent. This step allows the user to be “**topology independent**”. For example, let’s assume the user is working with a geometry where the upper surface is made up of two faces, and a week later the user gets a new geometry where the upper surface is made up of five faces; using the abstraction methodology allows the user to use the same blocking strategy on the different topologies [In the example above the upper surface has to be grouped using a sewn object].
2. Datum points. Geometric nodes or datum points need to be selected from the original geometry or the abstracted geometry. There are a number of point selection mechanisms that allow the user to select the points parametrically. So as the geometry changes the node coordinates change accordingly. The default point selection mechanism can be set by selecting one of the options under “**Default Assist Function**” from the pull down menu in the main sketcher form. The user can access more options using right mouse button before making the point selection (when working with graphics canvas a right mouse selection provides the user with options or terminates the current operation).
3. Mesh vertices. Are constructed either from the datum points or reference the geometry directly. The recommended method is to use the datum points.
4. Mesh edges. A Mesh edge is constructed from two mesh vertices.
5. Mesh faces. A mesh face is constructed from four mesh edges, or four mesh vertices selected in order.
6. Mesh blocks. A mesh block is constructed from six mesh faces or eight vertices selected in order.

2D vehicle centerline parametric geometry

A set of tools has been developed and implemented that allows IDEA to generate a three dimensional hypersonic vehicle using the keel line as a starting point. Using geometric reasoning, the system automatically generates a two dimensional centerline slice of the vehicle identifying critical geometric features that will be used to automatically build a two dimensional mesh topology. This example is illustrated in Figure 1.

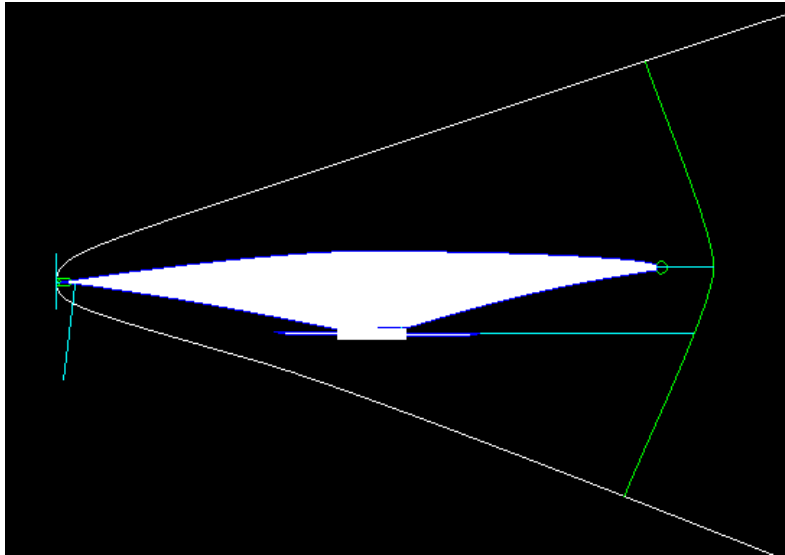


Figure 1. A two-dimensional slice and the building block for a parametric mesh topology.

Using the mesh topology depicted in Figure 1 and using the concept of geometric abstraction, the user can build more complex mesh topologies as illustrated in Figure 2. The mesh topology in Figure 2 was generated by Paul Ferlemann of the Hypersonic Air Breathing Propulsion Branch at NASA Langley.

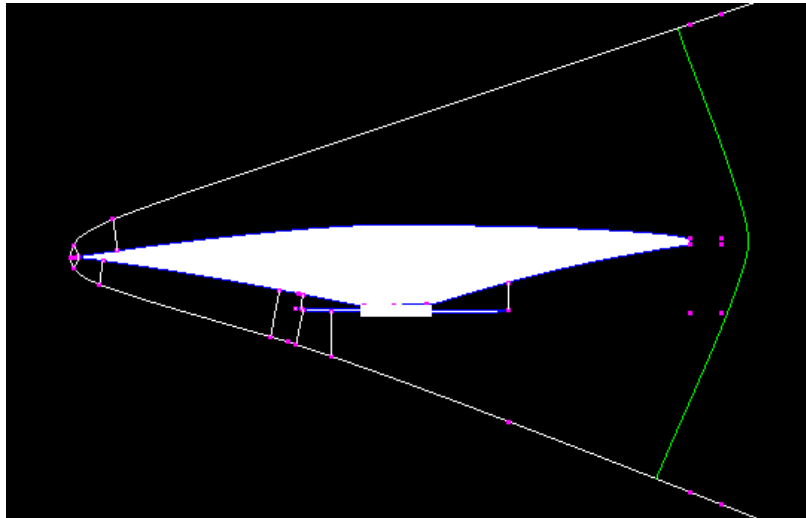


Figure 2. Additional mesh topology features are added by the user using the mesh topology module.

Using the mesh topology defined in Figure 2, the user can now associate mesh edges with the vertices, geometry, and the mesh topology curves to build the parametric structured mesh. Figure 3 illustrates the mesh edges that the user generated using the mesh topology defined in Figure 2.

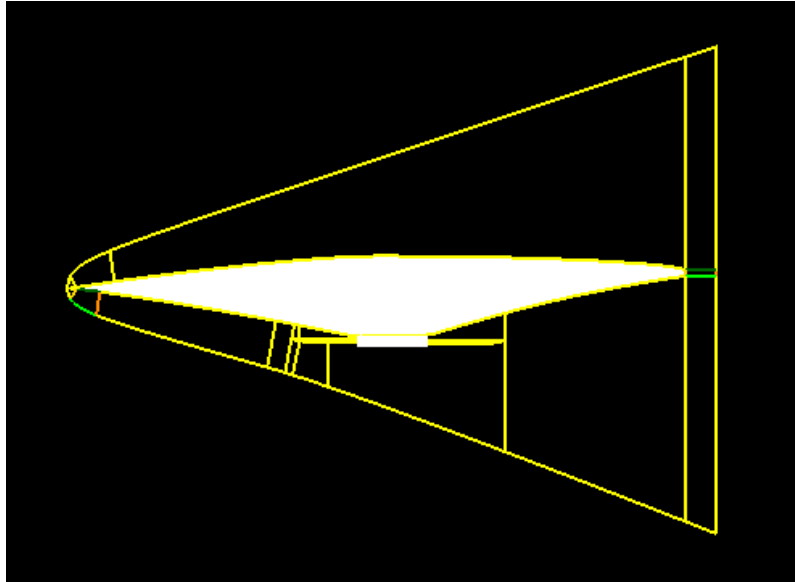


Figure 3. Mesh edges.

Modifying the cowl inlet flap angle from zero to four degrees results in the configuration changing as shown in Figure 4. The mesh edges update automatically without any user intervention.

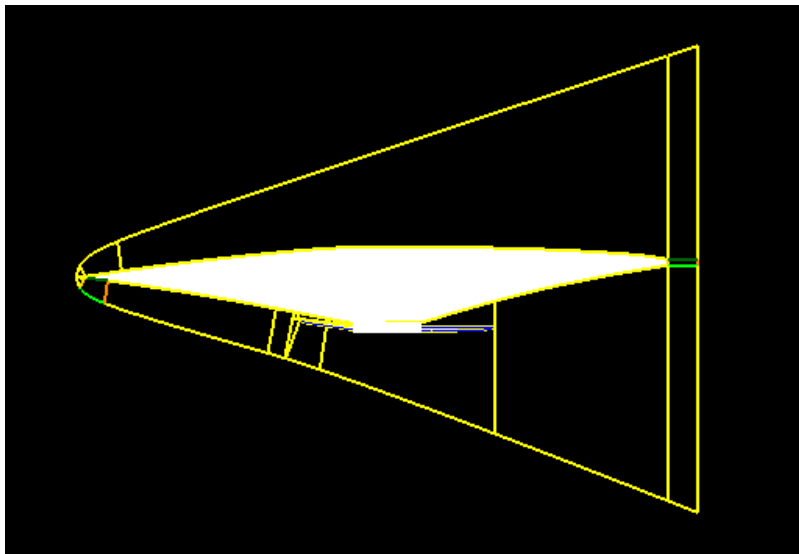


Figure 4. Inlet flap angle modified.

A mesh dimensioning module has been developed allowing the user to dimension the mesh using three different approaches that do not require the user to dimension each mesh edge separately. These methods are:

1. Select an edge dimension the edge, and every edge that belongs to the group is automatically dimensioned.
2. Dimension based on element sizes in the flow and across the flow.
3. Dimension based on a maximum acceptable element size.

Figure 5 illustrates the form that is used to access these methods. When dimensioning based on a maximum acceptable element size and using VULCAN for flowfield analysis, the system automatically re-dimensions the mesh to maintain the maximum element size criteria.

MESH EDGE DATA		
Mesh blocks container		mesh-topology-manager-class-0003
General Boundary Conds	Select/clear Boundary faces	Draw selected boundary faces
Order edge vertices based on boundaries		
Select mesh edge	Uniform distribution	
Distr params		
Number of elements		
Apply dist. to edge/Draw Ed faces	Reverse/Draw mesh edge	
Reset group elem quantity	Reverse group edge direction	
Apply/check group	Set group based on max elem size	
Draw/color mesh edge group	max element size	
Size elements across flow	0.5	
Size elements in flow	1.0	
Flow direction	(1.0 0.0 0.0)	
Coarsness level	No coarsness	
Reset all element quantity	Apply flow to dimension edges	
Copy dist spacing based on source	Set elem size across all groups	
Draw mesh edges	Draw mesh faces	Draw mesh blocks
List BC spec	BC groupings	Set/Draw Patches
Add/Action label CUT	Print grid&VULCAN / [Only V] files	

Figure 5. Mesh dimensioning form.

Figure 6 illustrates different meshes generated for different cowl positions.

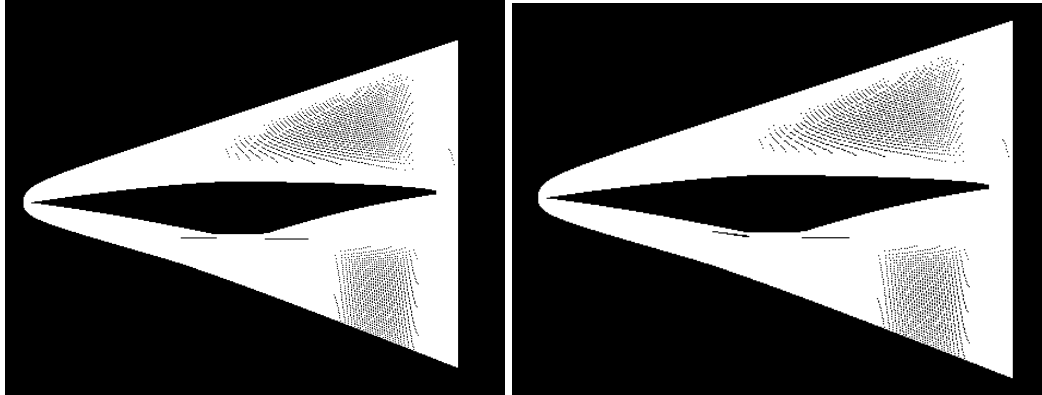


Figure 6. Mesh generated for two different cowl positions.

The team has developed a module that interfaces to the VULCAN analysis code automatically generating the different decks required by VULCAN. The mesh block interface information is automatically generated for VULCAN without requiring any user intervention. Figure 7 illustrates the top level of the VULCAN analysis graphical user interface.

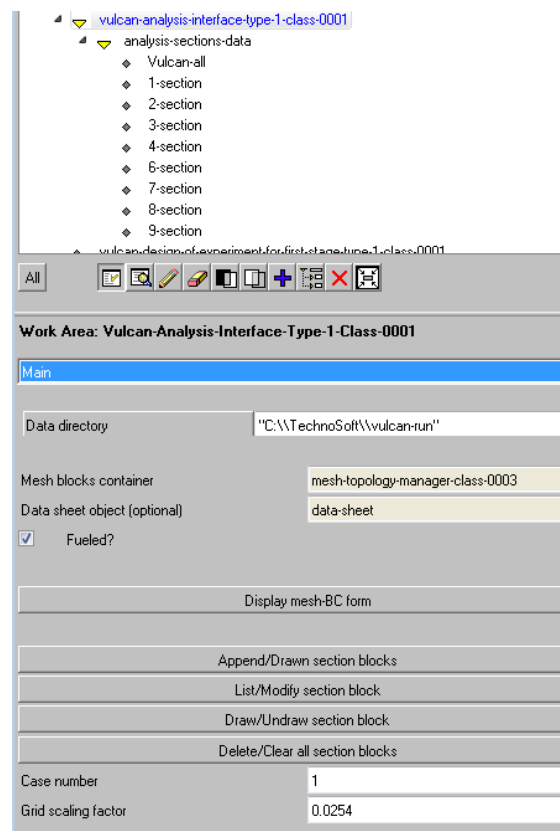


Figure 7. Top level VULCAN analysis graphical user interface.

The user can also group the different mesh blocks into sections. This process could be used when the problem at hand is such that the geometric changes do not affect the whole mesh, and user has an understanding of the behavior. To use this feature, the user will

need to define certain dependencies between sections using the graphical user interface illustrated in Figure 8.



Work Area: 1-section

Dependency

Set/Clear section dependencies

List section dependencies

Set/Clear independent vars dependencies

List independent vars dependencies

Draw/undraw section blocks

Draw/undraw section dependency blocks

Figure 8. Vulcan Section-Section dependency form.

Low speed flow path

A module has been developed to build a two dimensional (2D) and three dimensional (3D) representation of the low speed flow path. The 3D representation is used to generate the 2D flow path. Figure 8 illustrates the 3D jet engines modules. Figures 9 and 10 illustrate the 2D representation of this low speed flow path.

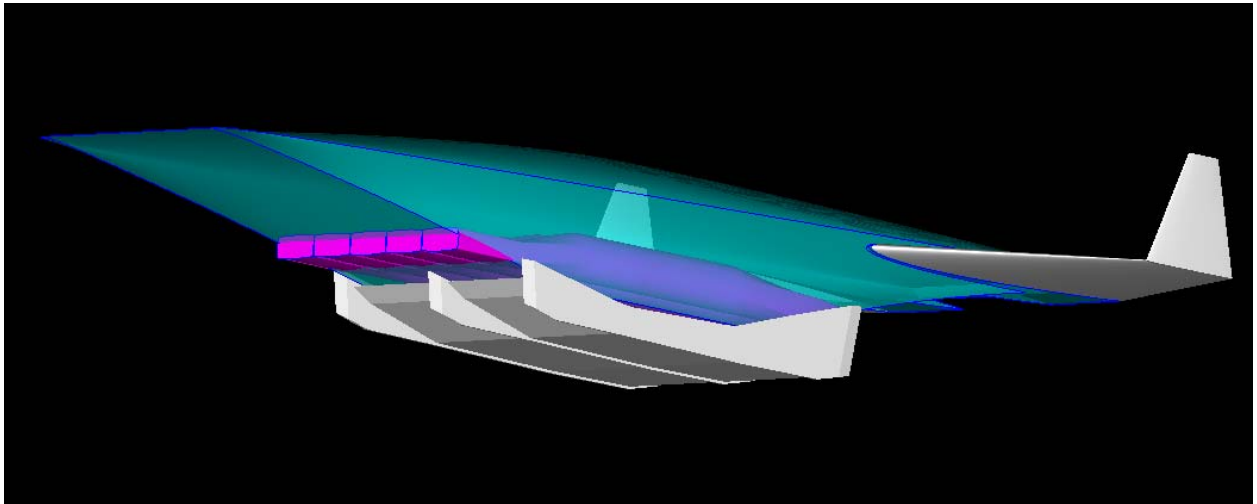


Figure 8. 3D jet engine module.

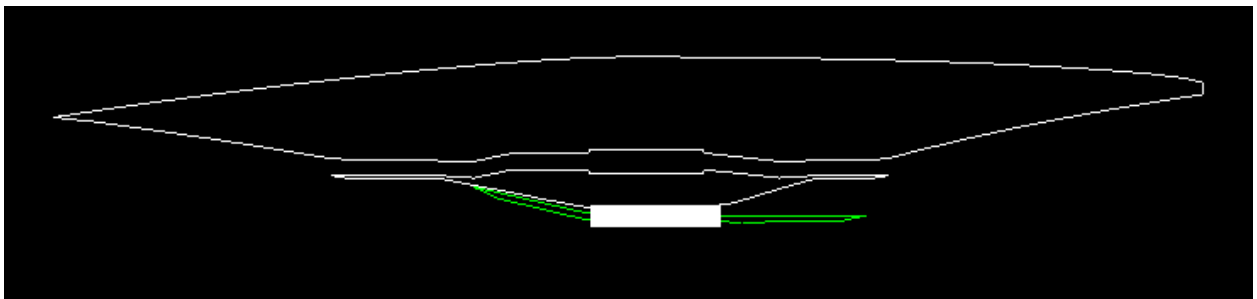


Figure 9. 2D slice with flaps at zero degrees.

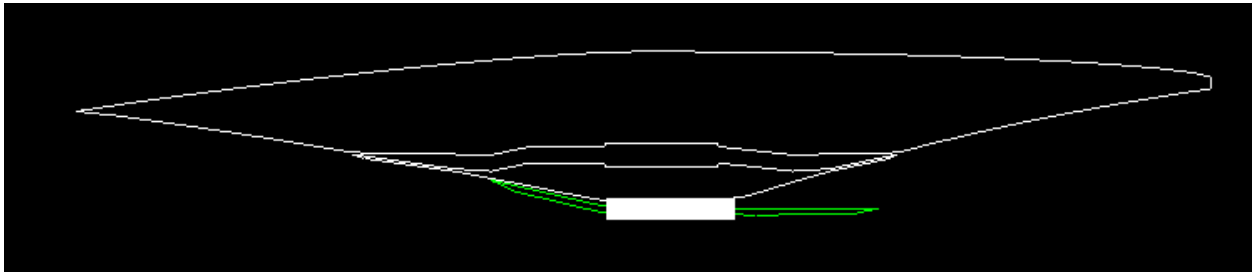


Figure 10. 2D slice with flaps closed.

The mesh topology for the over and under configuration has not been finalized.

REST inlet

A module has been developed allowing the user to build a geometric representation of a REST (Rectangular to Elliptical Shape Transition) inlet from a data deck generated by code developed by Dr. Rowan Gollan. Paul Ferlemann was instrumental in interpreting the data from the file and identifying a limited set of curves that can be used for defining a parametric geometric representation of the geometry that can be modified by the user without rerunning the code. Figure 11 shows the front view of REST inlet geometry. Figure 12 illustrates the side view of a 3D REST inlet. The inlet is made up of surface patches that are tangent continuous.

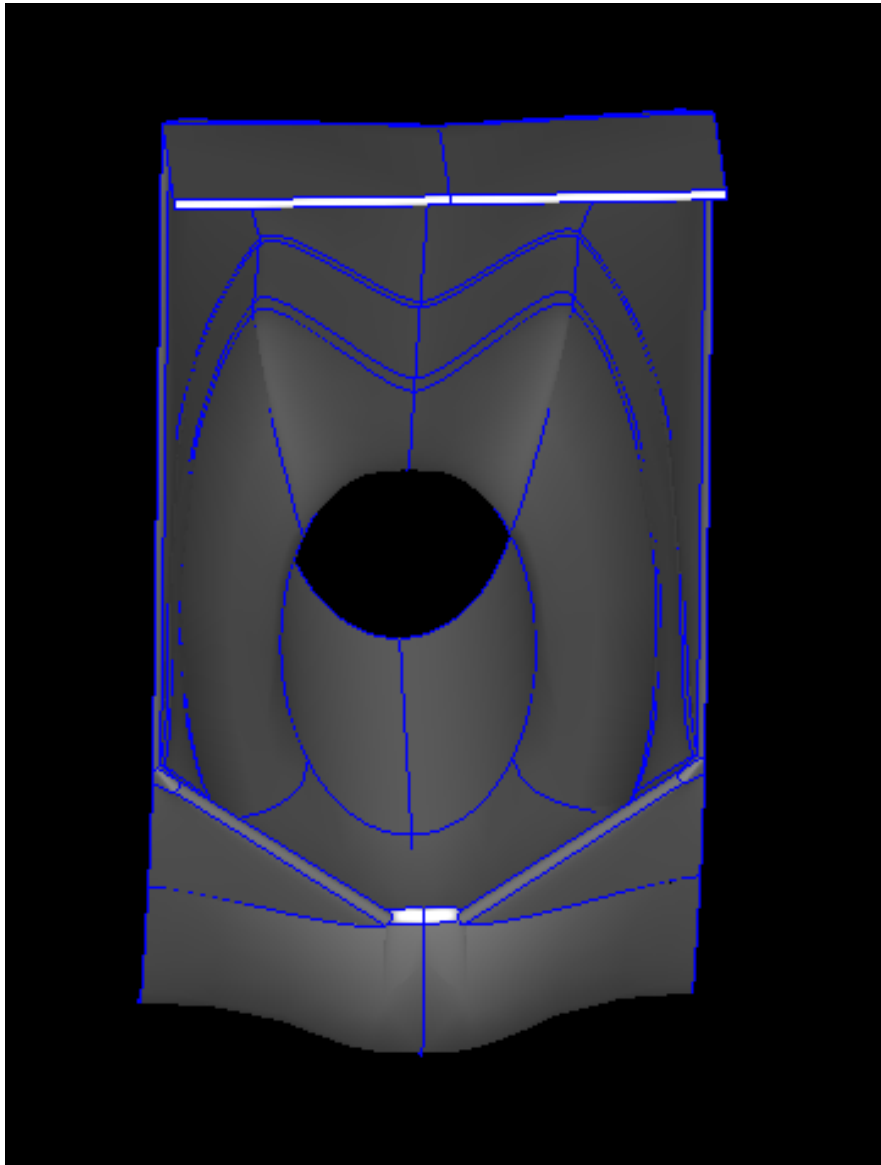


Figure 11. Front view of a 3D REST inlet parametric model.

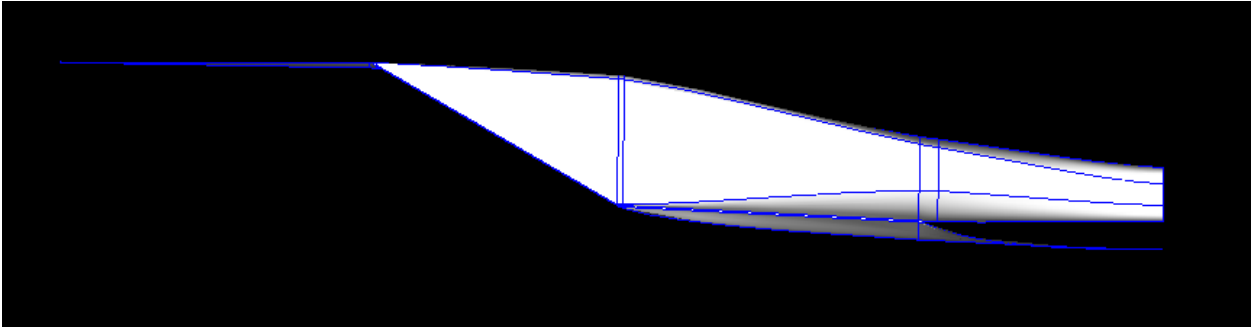


Figure 12. Side view of a 3D REST inlet.

A mesh topology was constructed using the 3D REST inlet geometry. The mesh has two thousand five hundred and eight faces; eight hundred and forty blocks. A lot of the faces and block were built from elemental faces, and blocks using mesh transition methodology. Figures 13, 14, 15, and 16 illustrate the different views of the constructed mesh topology.

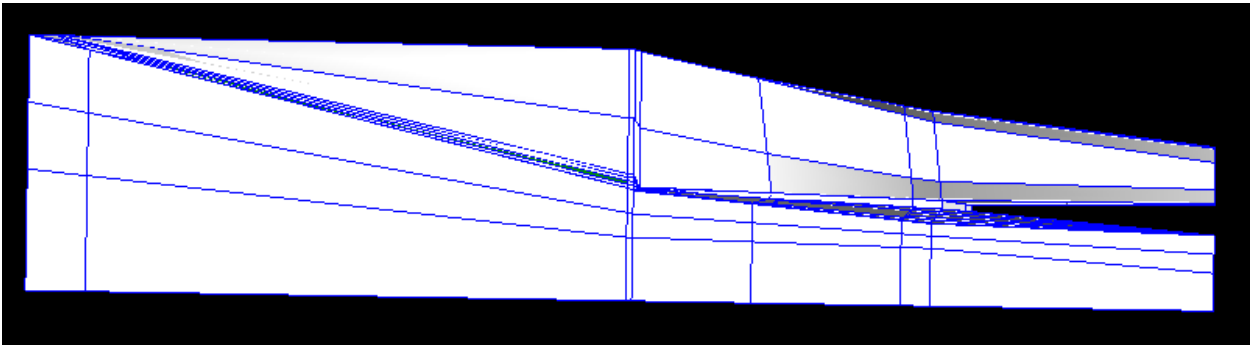


Figure 13. Side view 1 of the parametric mesh topology.

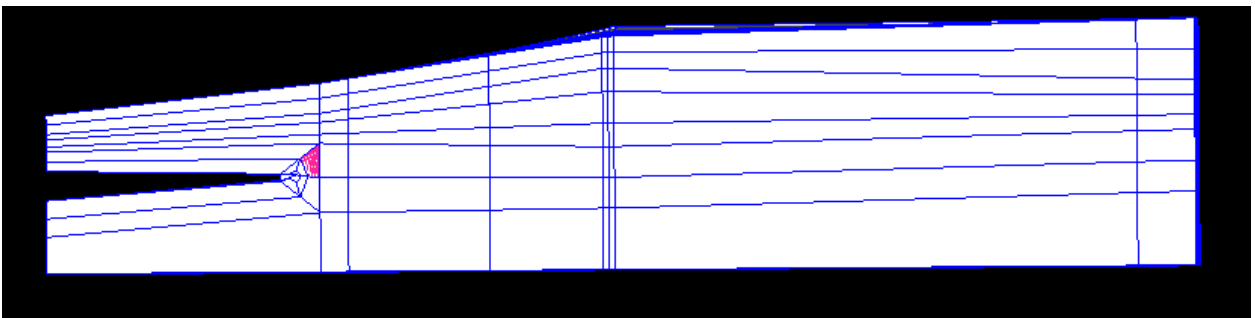


Figure 14. Side view 2 of the parametric mesh topology.

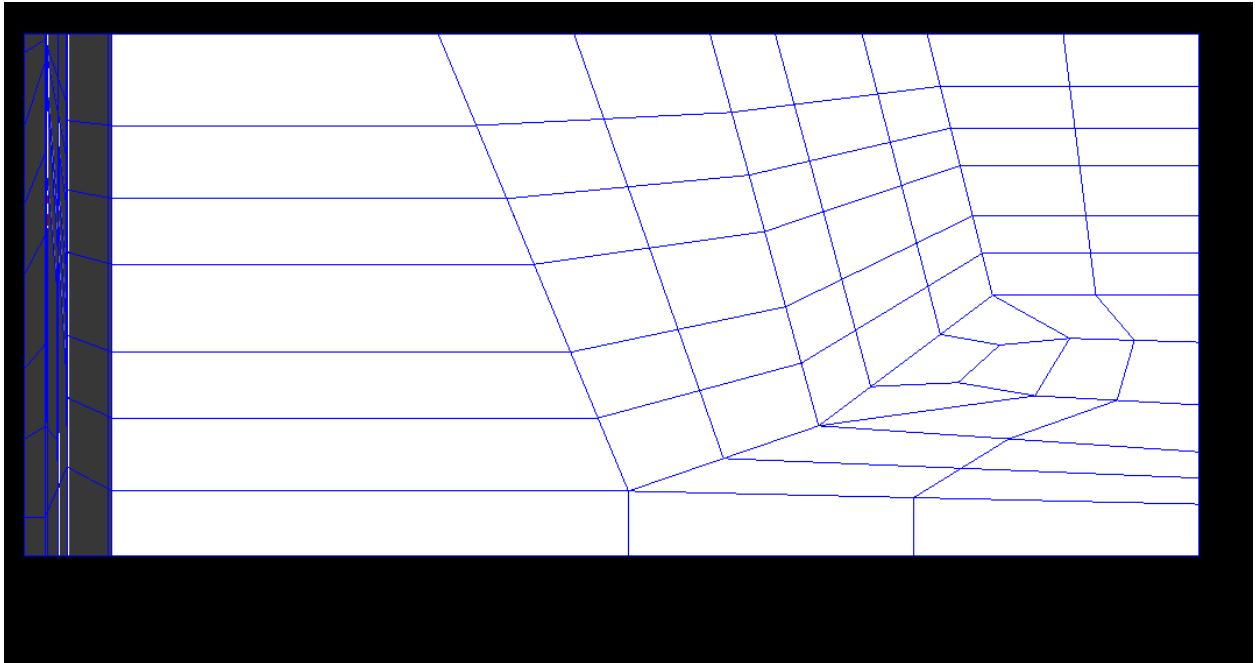


Figure 15. Front view of the parametric mesh topology.

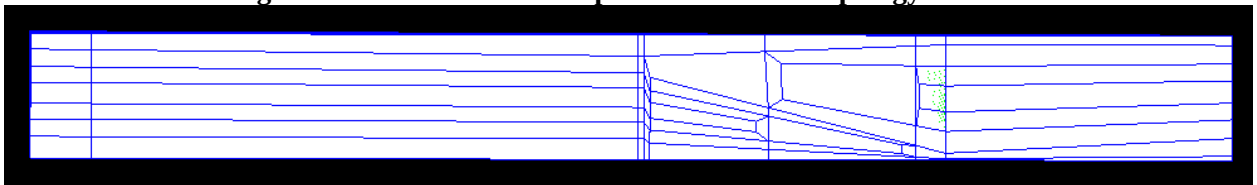


Figure 16. Bottom view of the parametric mesh topology.

Changes to the REST inlet geometry initiates the mesh topology module to update and reflect the changes in the geometry. The mesh can be dimensioned using the mesh dimensioning form illustrated in Figure 5.

Busemann inlet

A module has been designed and implemented to integrate a ray tracing code developed by Dr. Rob Baurle from the Hypersonics Air Breathing Propulsion Branch. The developed module allows the user to specify the input parameters and generate a 3D geometric representation of the inlet. Figure 1 illustrates the input parameters and the generated geometry. Internal and external inlet surfaces are blended to eliminate sharp corners. Figure 17 illustrates the Busemann inlet mesh geometry. Work has not been finalized on the mesh topology for this type of inlet.

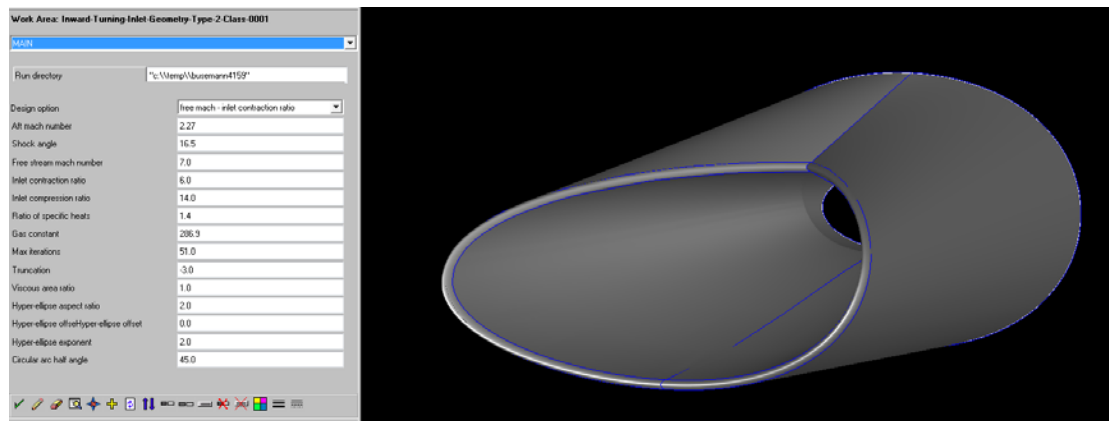


Figure 17. Busemann Inlet geometry.

Concluding remarks

This task has been instrumental in exploring the different requirements for separating the mesh topology from the geometric topology while maintaining the link between the mesh and the geometry. At this point, the developed methodology requires user inputs to identify and label the different geometric features. If the geometries are built using IDEA, and design intentions are clearly defined, then the tagging process can be omitted, and the user can proceed directly to mesh generation.

The developed modules are unique in that they allow the user to identify key features (e.g. combustor entrance, start of inlet, nozzle surfaces ...) and associate a specific mesh topology with certain geometric features.

The developed modules succeeded in capturing the meshing strategy for a class of vehicles regardless of the vehicle geometry or scale. The system has illustrated the viability of that concept and the time savings achieved from such a concept. The user can run a Design-of-Experiments study using VULCAN with the mesh updating as the vehicle geometry is changing without any user intervention.

More work is needed in the area of elliptical smoothing across mesh boundaries, refine the structured meshing graphical user interface, and exercise the mesh topology generator and mesher on different vehicle configurations.